

PN - JP2001209735 A 20010803  
PD - 2001-08-03  
PR - JP20000020375 20000128  
OPD - 2000-01-28  
TI - DATA PROCESSING WORKFLOW SYSTEM USING FILE EXPRESSED  
IN MARK-UP LANGUAGE  
IN - ITO KAZUHIKO; NISHIMURA AKIRA; KASUBUCHI KIYOTAKA  
PA - DAINIPPON SCREEN MFG  
IC - G06F19/00  
© WPI / DERWENT

TI - Data processing work flow system sets up and issues job information as a job  
ticket based on functional information  
PR - JP20000020375 20000128  
PN - JP2001209735 A 20010803 DW200161 G06F19/00 012pp  
PA - (DNIS ) DAINIPPON SCREEN SEIZO KK  
IC - G06F19/00  
AB - JP2001209735 NOVELTY - The system has a data processing station (13)  
which issues a job ticket indicating data processing content. The job information is set up and  
issued as a job ticket based on functional information. The functional information and the job  
ticket are text files expressed in a mark-up language.  
- USE - Data processing work flow system using file expressed in mark-up language.  
- ADVANTAGE - Simplifies data processing work flow process in different platforms.  
- DESCRIPTION OF DRAWING(S) - The figure shows the block diagram of data  
processing work flow system. (Drawing includes non-English language text).  
- Data processing station 13  
- (Dwg.1/7)  
OPD - 2000-01-28  
AN - 2001-545817 [61]  
© PAJ / JPO

PN - JP2001209735 A 20010803  
PD - 2001-08-03  
AP - JP20000020375 20000128  
IN - KASUBUCHI KIYOTAKA; ITO KAZUHIKO; NISHIMURA AKIRA  
PA - DAINIPPON SCREEN MFG CO LTD  
TI - DATA PROCESSING WORKFLOW SYSTEM USING FILE EXPRESSED  
IN MARK-UP LANGUAGE  
AB - PROBLEM TO BE SOLVED: To provide a data processing workflow system  
for easily realizing a data processing workflow even between different platforms.  
- SOLUTION: A data processing station 13 registers the information of functions  
executable by itself in a service provider 11. A client 12 sets job information based on the  
function information, and issues a job ticket. The function information and the job ticket are  
both a document file expressed with a mark-up document following common document type  
definition.  
I - G06F19/00

**THIS PAGE BLANK (USPTO)**

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号  
特開2001-209735  
(P2001-209735A)

(43) 公開日 平成13年8月3日 (2001.8.3)

(51) Int.Cl.<sup>7</sup>  
G 0 6 F 19/00

識別記号

F I  
G 0 6 F 15/22

テーマコード<sup>\*</sup> (参考)  
N

審査請求 未請求 請求項の数11 O L (全 12 頁)

(21) 出願番号 特願2000-20375 (P2000-20375)

(22) 出願日 平成12年1月28日 (2000.1.28)

(71) 出願人 000207551  
大日本スクリーン製造株式会社  
京都府京都市上京区堀川通寺之内上る4丁目天神北町1番地の1

(72) 発明者 粕淵 清孝  
京都市上京区堀川通寺之内上る4丁目天神北町1番地の1 大日本スクリーン製造株式会社内

(72) 発明者 伊藤 和彦  
京都市上京区堀川通寺之内上る4丁目天神北町1番地の1 大日本スクリーン製造株式会社内

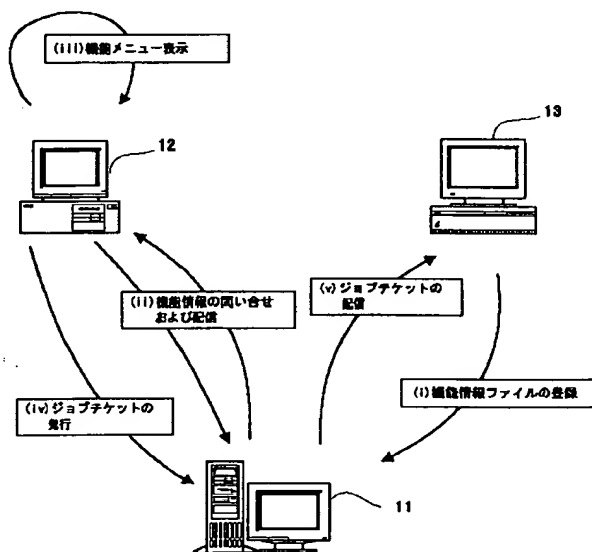
最終頁に続く

(54) 【発明の名称】 マークアップ言語で表現されたファイルを用いたデータ処理ワークフローシステム

(57) 【要約】

【課題】 異なるプラットフォーム間でも簡単にデータ処理ワークフローを実現することができるデータ処理ワークフローシステムを提供することを目的とする。

【解決手段】 データ処理ステーション13は、自身が実現可能な機能情報をサービスプロバイダ11に登録する。クライアント12は、機能情報に基づいてジョブ情報を設定するとともに、ジョブチケットを発行する。機能情報およびジョブチケットは、ともに共通の文書型定義に従うマークアップ文書で表現される文書ファイルである。



## 【特許請求の範囲】

【請求項1】 データ処理内容を表したジョブチケットを発行するジョブ発行ステーションと、前記データ処理内容に従って対象データを処理するデータ処理ステーションとが、ネットワーク接続されたデータ処理ワークフローシステムにおいて、

前記データ処理ステーションは、当該データ処理ステーションが実現可能な機能情報を前記ジョブ発行ステーションに提供する機能情報提供手段を備え、

前記ジョブ発行ステーションは、前記機能情報に基づいてジョブ情報を設定するジョブ設定手段と、設定されたジョブ情報をジョブチケットとして発行するジョブチケット発行手段とを備え、

前記機能情報および前記ジョブチケットは、共通の文書型定義に従うマークアップ言語で表現される文書ファイルであるデータ処理ワークフローシステム。

【請求項2】 前記ジョブ設定手段は、前記機能情報の内容を表示する機能情報表示手段と、表示された機能情報の内容に従ってジョブ情報を選択的に設定するジョブ選択手段からなる請求項1記載のデータ処理ワークフローシステム。

【請求項3】 前記共通の文書型定義を表す定義ファイルが、ジョブ発行ステーションおよびデータ処理ワークステーションからともに参照可能な記憶デバイスに保存されている請求項1記載のデータ処理ワークフローシステム。

【請求項4】 前記マークアップ言語は、XMLである請求項1乃至3記載のデータ処理ワークフローシステム。

【請求項5】 データ処理内容を表したジョブチケットを発行するクライアントと、前記データ処理内容に従って対象データを処理するデータ処理ステーションと、サービスプロバイダとが、ネットワーク接続されたデータ処理ワークフローシステムにおいて、

前記データ処理ステーションは、当該データ処理ステーションで実現可能な機能情報を前記サービスプロバイダに登録する機能情報登録手段を備え、

前記クライアントは、前記サービスプロバイダに登録された機能情報を参照し当該機能情報に基づいてジョブ情報を設定するジョブ設定手段と、設定されたジョブ情報をジョブチケットとして発行するジョブチケット発行手段とを備え、

前記サービスプロバイダは、前記機能情報および前記ジョブチケットをクライアントおよびデータ処理ステーションに配信する配信管理手段を備え、

前記機能情報および前記ジョブチケットは、共通の文書型定義に従うマークアップ言語で表現される文書ファイルであるデータ処理ワークフローシステム。

【請求項6】 前記サービスプロバイダは、前記データ処理ステーションにより登録された機能情報を、登録元

のデータ処理ステーションを識別する識別コードと対応付けて管理する登録管理手段を備え、

前記配信管理手段は、前記クライアントがジョブチケットを発行する際に参照した機能情報の登録元のデータ処理ステーションを、前記識別コードに基づいて特定し、当該特定されたデータ処理ステーションに前記ジョブチケットを配信する請求項5記載のデータ処理ワークフローシステム。

【請求項7】 前記配信管理手段は、ジョブチケットのジョブ情報を解析し、ジョブ情報として設定されたデータ処理内容が実現可能なデータ処理ステーションを検索し、検索されたデータ処理ステーションに前記ジョブチケットを配信する請求項5記載のデータ処理ワークフローシステム。

【請求項8】 前記配信管理手段は、検索されたデータ処理ステーションのなかから、データ処理効率において最適なデータ処理ステーションに前記ジョブチケットを配信する請求項7記載のデータ処理ワークフローシステム。

【請求項9】 前記ジョブ設定手段は、前記機能情報の内容を表示する機能情報表示手段と、表示された機能情報の内容に従ってジョブ情報を選択的に設定するジョブ選択手段からなる請求項5乃至8記載のデータ処理ワークフローシステム。

【請求項10】 前記共通の文書型定義を表す定義ファイルが、ジョブ発行ステーションおよびデータ処理ワークステーションからともに参照可能な前記サービスプロバイダに保存されている請求項5乃至8記載のデータ処理ワークフローシステム。

【請求項11】 前記マークアップ言語は、XMLである請求項5乃至8記載のデータ処理ワークフローシステム。

## 【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、マークアップ言語で表現されたファイルを用いたワークフローシステムに関する。

【0002】

【従来技術】新しいデータ処理ステーションが開発される毎に、それを利用するための専用のユーザーインターフェイスや、他の装置からネットワークを介して利用するための専用インターフェイスを作成している。例えば、あるクライアントコンピュータが、特定のデータ処理ステーションに対して、データ処理内容を表したジョブチケットを発行する場合、そのジョブチケットは、ジョブチケット発行先のデータ処理ステーション特有の規則に基づいて記述されたデータファイルである。データ処理ステーションは、この特有の規則に基づいてジョブチケットを解釈し、データ処理を実行する。

【0003】

【発明が解決しようとする課題】従来技術において、新しいデータ処理ステーションが開発される毎に、それを利用するための専用のユーザーインターフェイスや、他の装置からネットワークを介して利用するための専用インターフェイスを作成しなければならなかった。このため、開発時間や開発コストが必要となり、結果として、製品のコストアップを招いていた。また、プラットフォーム、すなわちオペレーティングシステムが異なるコンピュータやワークステーション間では、データ交換する際の制約が多くなり、そのため、インターフェースの作成には、多大な時間およびコストを必要とする。

【0004】本発明は、上記問題点を解決し、異なるプラットフォーム間でも簡単にデータ処理ワークフローを実現することができるデータ処理ワークフローシステムを提供することを目的とする。

【0005】

【課題を解決するための手段およびその作用・効果】前述した課題の少なくとも一部を解決するための手段として、以下に示す構成をとった。請求項1記載の発明は、データ処理内容を表したジョブチケットを発行するジョブ発行ステーションと、前記データ処理内容に従って対象データを処理するデータ処理ステーションとが、ネットワーク接続されたデータ処理ワークフローシステムにおいて、前記データ処理ステーションは、当該データ処理ステーションが実現可能な機能情報を前記ジョブ発行ステーションに提供する機能情報提供手段を備え、前記ジョブ発行ステーションは、前記機能情報に基づいてジョブ情報を設定するジョブ設定手段と、設定されたジョブ情報をジョブチケットとして発行するジョブチケット発行手段とを備え、前記機能情報および前記ジョブチケットは、共通の文書型定義に従うマークアップ言語で表現される文書ファイルであることを特徴とする。

【0006】この発明によれば、データ処理ステーションから提供される機能情報と、ジョブ発行ステーションから発行されるジョブチケットは、ともに、マークアップ言語で表現される文書ファイルで構成され、さらに、共通の文書型定義に従っているため、データ処理ステーションやジョブ発行ステーションが動作するオペレーティングシステムに関係なく、簡単にデータ交換が可能となる。また、両者間の特別なインターフェイスを設ける必要もない。

【0007】請求項2に記載の発明は、請求項1記載の発明において、前記ジョブ設定手段は、前記機能情報の内容を表示する機能情報表示手段と、表示された機能情報の内容に従ってジョブ情報を選択的に設定するジョブ選択手段からなることを特徴とする。

【0008】この発明によれば、データ処理ステーションから提供された機能情報を表示し、その表示の内容に従ってジョブ情報を選択的に設定するので、簡単な操作でジョブ情報を設定することができる。

【0009】請求項3に記載の発明は、請求項1記載の発明において、前記共通の文書型定義を表す定義ファイルが、ジョブ発行ステーションおよびデータ処理ワークステーションからともに参照可能な記憶デバイスに保存されていることを特徴とする。

【0010】この発明によれば、文書型定義を表す定義ファイルが、ジョブ発行ステーションおよびデータ処理ワークステーションからともに参照可能な記憶デバイスに保存されているので、両ステーションは、定義ファイルを参照してマークアップ言語で表現される文書ファイルを解釈することができる。

【0011】請求項4に記載の発明は、請求項1乃至3記載の発明において、前記マークアップ言語は、XMLであることを特徴とする。

【0012】この発明によれば、マークアップ言語としてXMLを用いているので、機能情報およびジョブチケットの文書構造を簡単に拡張することができる。

【0013】請求項5に記載の発明は、データ処理内容を表したジョブチケットを発行するクライアントと、前記データ処理内容に従って対象データを処理するデータ処理ステーションと、サービスプロバイダとが、ネットワーク接続されたデータ処理ワークフローシステムにおいて、前記データ処理ステーションは、当該データ処理ステーションで実現可能な機能情報を前記サービスプロバイダに登録する機能情報登録手段を備え、前記クライアントは、前記サービスプロバイダに登録された機能情報を参照し当該機能情報に基づいてジョブ情報を設定するジョブ設定手段と、設定されたジョブ情報をジョブチケットとして発行するジョブチケット発行手段とを備え、前記サービスプロバイダは、前記機能情報および前記ジョブチケットをクライアントおよびデータ処理ステーションに配信する配信管理手段を備え、前記機能情報および前記ジョブチケットは、共通の文書型定義に従うマークアップ言語で表現される文書ファイルであることを特徴とする。

【0014】この発明によれば、データ処理ステーションから登録される機能情報と、ジョブ発行ステーションから発行されるジョブチケットを、サービスプロバイダで配信管理する。また、機能情報およびジョブチケットは、ともに、マークアップ言語で表現される文書ファイルで構成され、さらに、共通の文書型定義に従っている。このため、データ処理ステーションやクライアントが動作するオペレーティングシステムに関係なく、ネットワーク上で、簡単にデータ交換が可能となる。また、ネットワーク間のデータ転送のため特別なインターフェイスを設ける必要もない。

【0015】請求項6に記載の発明は、請求項5記載の発明において、前記サービスプロバイダは、前記データ処理ステーションにより登録された機能情報を、登録元のデータ処理ステーションを識別する識別コードと対応

付けて管理する登録管理手段を備え、前記配信管理手段は、前記クライアントがジョブチケットを発行する際に参照した機能情報の登録元のデータ処理ステーションを、前記識別コードに基づいて特定し、当該特定されたデータ処理ステーションに前記ジョブチケットを配信することを特徴とする。

【0016】この発明によれば、サービスプロバイダが、データ処理ステーションの機能情報を登録管理しているので、クライアントは、ジョブチケットの発行先に関する情報を入力する必要はなく、このためクライアントオペレータの操作性が向上する。

【0017】請求項7に記載の発明は、請求項5記載の発明において、前記配信管理手段は、ジョブチケットのジョブ情報を解析し、ジョブ情報として設定されたデータ処理内容が実現可能なデータ処理ステーションを検索し、検索されたデータ処理ステーションに前記ジョブチケットを配信することを特徴とする。

【0018】この発明によれば、サービスプロバイダが、ジョブチケットのジョブ情報を解析して、その解析結果に基づいて、ジョブチケットの配信先を決定し、ジョブチケットを配信するので、クライアントは、ジョブチケットの配信先に関する情報を入力する必要はなく、このためクライアントオペレータの操作性が向上する。

【0019】請求項8に記載の発明は、請求項7記載の発明において、前記配信管理手段は、検索されたデータ処理ステーションのなかから、データ処理効率において最適なデータ処理ステーションに前記ジョブチケットを配信することを特徴とする。

【0020】この発明によれば、サービスプロバイダが、データ処理効率の最適なデータ処理ステーションにジョブチケットを配信するので、ネットワーク上のトータルシステムにおいてデータ処理効率が向上する。

【0021】請求項9に記載の発明は、請求項5乃至8に記載の発明において、前記ジョブ設定手段は、前記機能情報の内容を表示する機能情報表示手段と、表示された機能情報の内容に従ってジョブ情報を選択的に設定するジョブ選択手段からなることを特徴とする。

【0022】この発明によれば、データ処理ステーションから提供された機能情報を表示し、その表示の内容に従ってジョブ情報を選択的に設定するので、簡単な操作でジョブ情報を設定することができる。

【0023】請求項10に記載の発明は、請求項5乃至8に記載の発明において、前記共通の文書型定義を表す定義ファイルが、ジョブ発行ステーションおよびデータ処理ワークステーションからともに参照可能な前記サービスプロバイダに保存されていることを特徴とする。

【0024】この発明によれば、文書型定義を表す定義ファイルが、クライアントおよびデータ処理ワークステーションからともに参照可能なサービスプロバイダに保存されているので、両者は、定義ファイルを参照してマ

ークアップ言語で表現される文書ファイルを解釈することができる。

【0025】請求項11に記載の発明は、請求項5乃至8に記載の発明において、前記マークアップ言語は、XMLであることを特徴とする。

【0026】この発明によれば、マークアップ言語としてXML言語を用いているので、機能情報およびジョブチケットの文書構造を簡単に拡張することができる。

【0027】

【発明の実施の形態】〔0. マークアップ言語と文書型定義〕具体的な本発明の実施形態を説明する前に、本発明に係る一般的なマークアップ言語および文書型定義について説明する。マークアップ言語とは、文書中にタグをつけて、文書構造や修飾情報などを追加するための言語で、構造化文書を作成することができる。一般的に、タグには、開始タグと終了タグがあり、原則としてこれらのタグは一対一に対応する。対応する開始タグと終了タグの間に、そのタグに対応する要素の内容が記述される。また、対応する開始タグから終了タグまでの要素は、さらにその中に別の要素を含むことができる。このため、マークアップ言語を用いると、階層的な論理構造を有するデータベースを構成することも可能となる。

【0028】また、マークアップ言語を利用した文書の特徴は、データがあくまでテキスト形式であり、プラットフォーム、すなわち各ワークステーションやコンピュータのオペレーティングシステムに依存することがない。そのため、種々のプラットフォーム間同士のデータ交換を容易に実現することができる。

【0029】マークアップ言語の代表的な存在としては、SGML(Standard Generalized Markup Language)、XML(eXtensible Markup Language)およびHTML(Hyper Text Markup Language)などがある。これら言語に基づいて作成された文書は、SGML文書、XML文書、またはHTML文書と呼ばれる。ここでは、マークアップ言語で作成された文書のことを「マークアップ言語文書」、それを表す電子的なファイルを「マークアップ言語文書ファイル」とそれぞれ呼ぶことにする。

【0030】また、文書型定義とは、マークアップ言語文書でどのようなタグを使用することができるのか、そのタグの要素に対してどのような属性を使用することができるのか、各要素はどのようなデータ型なのか等のマークアップ言語文書の型を定義することである。一般的にSGMLやXMLでは、DTDと呼ばれる文書型定義がある。また、XMLの場合は、さらにXML NamespaceやXML Schemaと呼ばれる、DTDをより拡張し、発展させた文書型定義もある。

【0031】〔1-1. データ処理ワークフローシステムの構成〕本発明の実施の形態を実施例に基づき説明する。図1は、本発明に係るデータ処理ワークフローシステムの構成図である。データ処理ワークフローシステム

1は、サービスプロバイダ11と、複数のクライアント12と、複数のデータ処理ステーション13が、ネットワーク14で接続されている。サービスプロバイダ11は、CPU、メモリ、ハードディスク、モニタ、キーボードおよびマウスから構成されるワークステーションで、オペレーティングシステムソフトウェアおよびサーバプログラムにより動作するいわゆるサーバである。具体的には、WWWサーバなどに相当する。クライアント12は、CPU、メモリ、ハードディスク、モニタ、キーボードおよびマウスから構成されるコンピュータで、オペレーティングシステムソフトウェアおよびクライアントプログラムにより動作するいわゆるクライアントコンピュータである。具体的には、WWWクライアントに相当する。

【0032】データ処理ステーション13は、各種のデータ処理を行う装置やステーションである。例えば、印刷・製版の技術分野であれば、写真等の画像を読み取る画像入力装置、多数の画像データを蓄積・管理する画像データベース、ページ記述言語で表現されたページデータをラスターライズ処理するRIP装置、カラー画像の色変換処理を行うカラーマネージメント装置、各種データのネットワーク転送を行うネットワーク転送装置などである。なお、これらのデータ処理ステーション13には、その一部に少なくともコンピュータが含まれており、画像入力装置のような場合は、コンピュータと画像

入力するためのスキャナとが接続されたシステムの装置である。

【0033】ネットワーク14は、httpプロトコルに従ってデータ転送するもので、例としては、LAN、WAN、インターネット、イントラネット若しくはエクストラネットなどが挙げられる。

【0034】このように構成されたデータ処理ワークフローシステムでは、マークアップ言語文書ファイルを用いて各種データ交換が可能である。特に、本発明では、各データ処理ステーションが実現可能な機能情報およびクライアント11からサービスプロバイダ12またはデータ処理ステーション13に発行または配信されるジョブチケットが、マークアップ言語文書で記述され、その文書ファイルの形式でデータ交換が行われる。なお、以下、この実施の形態においては、マークアップ言語文書として、XMLを採用した場合について説明する。

【0035】【文書型定義を行うための定義ファイル】XMLやSGMLなどのマークアップ言語などで表現される文書の構造を定義するために、通常DTD (Document Type Definition) と呼ばれる文書型定義が用いられる。XMLの場合に適用される文書型定義の例を表1に示す。

【0036】

【表1】

#### 定義ファイルDFM ファイル名:XMLTicket.DTD

```

1      <!-- Ticket Main DTD -->
2
3      <!ELEMENT Ticket (Kind, Function*)>
4      <!ELEMENT Kind (#PCDATA)>
5      <!ELEMENT Function (Item)*>
6      <!ELEMENT Item (#PCDATA)>
7
8      <!ATTLIST Function Name CDATA #REQUIRED>
```

【0037】この例においては、次のように各要素が階層的に定義されている。まず、要素型「Ticket」に属する要素が含まれ、さらに、この「Ticket」に属する要素には、要素型「Kind」に属する要素および要素型「Function」に属する要素の2つの要素が含まれる。このうち、要素型「Function」に属する要素には、さらに、要素型「Item」に属する要素を含む。なお、要素型「Function」に属する要素は、属性「Name」を持ちうる。なお、ここで、要素型「Function」に属する要素および要素型「Item」に属する要素は、「0」回以上の繰返し可能である。このように、表1の文書型定義は、図2に示したツリー構造（階層構造）で表すことができる。

【0038】なお、この文書型定義の定義ファイルDFMは、例えばファイル名「XMLTicket.DTD」として、サービスプロバイダ12に保存されており、各クライアン

ト11やデータ処理ステーション13から、URL (Uniform Resource Locator) やネットワークパスを設定することにより参照可能に構成されている。

【0039】なお、本発明の実施の形態において、もう1つの文書型の定義ファイルDFSが作成されている。この定義ファイルDFSは、表1で示した定義ファイルDFMの定義を継承して、さらに詳細な文書型を定義するファイルである。即ち、定義ファイルDFMは、主要部分の文書型を定義したいいわゆるメインの定義ファイルで、一方、定義ファイルDFSは、細部の文書型を定義したいいわゆるサブの定義ファイルである。表2に示されたサブ文書定義DFSの例を示す。

【0040】

【表2】

## 定義ファイルDFS ファイル名:XMLTicketSub.DTD

```

1      <!-- Ticket Sub DTD -->
2
3      <!ENTITY % XMLTicket PUBLIC "Ticket Main DTD"
4          "http://www...../XMLTicket.dtd">
5
6      <!ATTLIST Ticket ID (Ticket | Menu) "Ticket">
7      <!ATTLIST Function Select (1 | 1ge) "1">
8      <!ATTLIST Function Display (SelectBox | RadioButton) "SelectBox">
9
10     %XMLTicket;

```

【0041】この例で定義された文書型は、要素「Ticket」は、属性「ID」の値を持つことができ、その値は、「Ticket」または「Menu」のいずれかの値を取り、デフォルトとして、「Ticket」が選択される。属性「ID」が、「Ticket」の場合は、要素「Ticket」はジョブチケットの内容であることを示し、「Menu」の場合は、データ処理ステーションの機能情報の内容であることを示している。同様に、要素「Function」は、属性「Select」の値を持つことができ、その値は、「1」または「1ge」のいずれかの値を取り、デフォルトとして、「1」が選択される。「1」の場合は、1つのパターンのみ選択できることを示し、「1ge」の場合は、1以上のパターンを選択できることを示している。また、要素「Function」は、属性「Display」の値を持つことができ、その値は、「SelectBox」または「RadioButton」のいずれかの値を取り、デフォルトとして、「SelectBox」が選択される。「RadioButton」の場合は、メニューがラジオボタンで表示されることを示し、「SelectBox」の場合は、メニューがセレクトボックスで表示されることを示している。また、1行目では、定義ファイルDFMで規定された文書型定義ファイル「XMLTicket.DTD」を継承することも定義されている。

【0042】このように、定義ファイルDFSで設定された文書型定義をツリー構造（階層構造）で示すと図3のようになる。なお、この文書型定義の定義ファイルDFSも、メインの定義ファイルDFMと同様に、例えばファイル名「XMLTicketSub.DTD」として、サービスプロバイダ12に保存されており、各クライアント11やデータ処理ステーション13から、URL(Uniform Resource Locator)やネットワークパスを設定することにより参照可能に構成されている。

【0043】[1-2. ワークフロー]図4に、マークアップ言語文書ファイルを用いたデータ処理ワークフローの概略を示す。このデータ処理ワークフローの工程は、大きく分けて次の5つの工程からなる。

(i) 各データ処理ステーション13が、それぞれの機能情報ファイルをサービスプロバイダ11に登録する。  
 (ii) クライアント12が、サービスプロバイダ11に対して利用したい機能についての問い合わせを行い、これに対して、サービスプロバイダ11が、該当する機能情報ファイルをクライアント12に配信する。  
 (iii) 取得した機能情報ファイルをメニュー化して表示する。

(iv) メニュー化された機能情報に基づいてジョブ情報を設定し、サービスプロバイダ11に対してジョブチケットを発行する。

(v) サービスプロバイダ11は、クライアント12から受け取ったジョブチケットを解釈し、そのジョブを実行することのできる最適なデータ処理ステーション13に対して、そのジョブチケットを配信する。続いて、これらの5つの各工程の詳細について、説明する。

【0044】[(i) 機能情報ファイルの登録]データ処理ステーション13は、そのデータ処理ステーション13が実現可能な機能情報をサービスプロバイダ11に登録する。機能情報は、文書型定義の定義ファイルDFM(定義ファイルDFMのみ、若しくは定義ファイルDFMおよびDFS)に従ったマークアップ言語表現されたマークアップ言語文書で表される。マークアップ言語文書の一般的な例を表3に示す。

【0045】

【表3】



## 機能情報ファイル

```

1      <?xml version="1.0"?>
2
3      <!DOCTYPE XMLMenu SYSTEM
         "http://www...../XMLTicketSub.dtd">
4
5      <Ticket ID="Menu">
6        <Kind>A</Kind>
7        <Function Name="X" Display="RadioButton">
8          <Item>1</Item>
9          <Item>2</Item>
10         <Item>3</Item>
11        </Function>
12        <Function Name="Y" Select="1ge">
13          <Item>one</Item>
14          <Item>two</Item>
15          <Item>three</Item>
16        </Function>
17      </Ticket>

```

【0046】このマークアップ言語文書の1行目で、XMLで記述された文書であることを宣言し、3行目で、このマークアップ言語文書（XML文書）は、定義ファイルDFS（ファイル名「XMLMenuSub.DTD」）に基づいた文書型定義に従った文書であることを示し、5行以降で、各要素の内容を記述している。なお、この機能情報は、図5に示されたツリー構造で表すこともできる。

【0047】表3で表される機能情報の詳細について説明する。まず、「Ticket」の「ID」が、値「Menu」に設定されている。これは、要素「Ticket」の要素内容が、機能情報であることを示している。仮に、「ID」の値が「Ticket」であれば、ジョブチケットを表していることになる。要素「KIND」として「A」が設定されている。例えば、要素「KIND」は、データ処理ステーション

の種類を示すものである。さらに、この機能情報によれば、「X」機能と「Y」機能の2つの機能を備えている。「X」機能は、オプションとして、3つのパターン「1」、「2」、「3」から1つのものを選択することができ、ラジオボタンを用いてメニュー表示される。

「Y」機能は、オプションとして、3つのパターン「one」、「two」、「three」から1またはそれ以上を選択することができ、セレクトボックスを用いてメニュー表示される。

【0048】また、データ処理ステーション13がRIP装置である場合の、具体的な機能情報ファイルの例を表4に示す。

【0049】

【表4】

## 機能情報ファイル

```

1      <?xml version="1.0"?>
2
3      <!DOCTYPE XMLMenu SYSTEM
         "http://www...../XMLTicketSub.dtd">
4
5      <Ticket ID="Menu">
6        <Kind>RIP</Kind>
7        <Function Name="LPI" Display="RadioButton">
8          <Item>200</Item>
9          <Item>400</Item>
10         <Item>600</Item>
11        </Function>
12        <Function Name="Screen" Select="1ge">
13          <Item>FM</Item>
14          <Item>Square</Item>
15          <Item>Round</Item>
16          <Item>Chain</Item>
17        </Function>
18      </Ticket>

```

【0050】このマークアップ言語文書の1行目で、XMLで記述された文書であることを宣言し、3行目で、

このマークアップ言語文書（XML文書）は、定義ファイルDFS（ファイル名「XMLMenuSub.DTD」）に基づい

た文書型定義に従った文書であることを示し、5行以降で、各要素の内容を記述している。

【0051】表4で表される機能情報の詳細について説明する。まず、「Ticket」の「ID」が、値「Menu」に設定されている。これは、要素「Ticket」の要素内容が、機能情報であることを示している。要素「KIND」として「RIP」が設定されている。すなわち、この装置は、RIP装置であることを示している。さらに、「LPI」機能と「Screen」機能の2つの機能を備えている。「LPI」機能は、この装置により出力されるラスタライズデータの解像度を「200線」、「400線」、「600線」のなかから選択できる。「Screen」機能は、この装置により出力されるラスタライズデータの網点形状を「FM」、「Square」、「Round」、「Chain」のなかから選択できる。これらの各機能メニューは、ラジオボタンで表示されることが記述されている。この機能情報は、図6に示したツリー構造で表すこともできる。

【0052】サービスプロバイダ11は、機能情報ファイルとともに、その機能情報ファイルの登録元のデータ処理ステーション13のIPアドレスと対応付けて登録管理している。このIPアドレスは、登録元のデータ処理ステーション13を識別するために用いられる。もちろん、ネットワーク14に接続されている他のデータ処理ステーション13からの機能情報も登録される。

【0053】このように、サービスプロバイダ11のハードディスクには、マークアップ言語で表現され、かつ予め設定された文書型定義に従う形式で各データ処理ステーション13の機能情報ファイルがデータベースとして蓄積される。このため、特定の機能を有するデータ処理ステーション13がネットワーク上に存在するかどうかを高速に検索することができる。例えば、ネットワーク14に利用可能なRIP装置が接続されているかどうかを検索する場合、開始タグ<KIND>と終了タグ</KIND>で挟まれた要素「KIND」の要素の内容が文字列「RIP」である機能情報ファイルを検索すればよい。

【0054】なお、機能情報ファイルを登録するタイミングは、データ処理ステーション13のオペレータからの指示により行ってもよいし、また、データ処理ステーション13がネットワーク14上に接続されたときに自動登録するように構成してもよい。なお、自動登録する場合には、ネットワークに接続されたかどうかを検出するモジュールが各データ処理ステーション13に予めインストールされている必要がある。

【0055】[(ii) 機能情報の取得]クライアント12は、データ処理に利用したいデータ処理ステーション13に関する機能情報を取得するため、サービスプロバイダ11に利用したいデータ処理の機能についての問い合わせを行う。例えば、クライアント12は、ネットワークで利用可能なデータ処理ステーションのなかに、FM

網点を出力することが可能なRIP装置が接続されているかどうかの問い合わせを、サービスプロバイダ11に対して行う。この問い合わせに対して、サービスプロバイダ11は、登録されている機能情報ファイルのなかに、要素「KIND」に文字列「RIP」を含み、かつ、属性「Name」が「Screen」である要素「Function」に文字列「FM」を含む機能情報ファイルを抽出し、問い合わせのあったクライアント12に配信する。

[(iii) メニュー表示]クライアント12は、サービスプロバイダ11から配信された機能情報ファイルを受け取る。機能情報ファイルは、クライアント12上で動作する表示用アプリケーションによって、クライアント12のモニタにメニュー形式で表示する。機能メニューを表示するための表示用モジュール(アプリケーション)は、まず、XML文書を解析し、属性を持った要素のツリー構造を作成する。これは、XMLパーサとも呼ばれ、参照する文書型定義に対して適正形式であるかどうかの検証も行われる。

【0056】続いて、ツリー構造に解剖された各要素を予め設定された表示ルールに基づいて、メニュー表示する。図7に、表4で示された機能情報ファイルをメニュー表示した例を示す。

【0057】機能メニューは、ウインドウ20内に表示される。ウインドウ20内には、データ処理ステーションの種類を示す文字21、2つのオプションボックス23および25、2つのコマンドボタン26および27が表示される。文字21の内容は、データ処理ステーションの機能を表し、「RIP」と記述される。オプションボックス23は、「LPI」機能において選択可能な3つのパターン「200線」、「400線」、「600線」が表示され、このうち1つのパターンをマウスをクリックすることにより選択することができる。オプションボックス25は、「Screen」機能において選択可能な4つのパターン「FM」、「Square」、「Round」、「Chain」が表示され、このうち1つのパターンをマウスをクリックすることにより選択することができる。なお、オプションボックス23および25には、その機能名が文字22、24としてそれぞれ表示されている。コマンドボタン26は、オプションボタンを選択することによって設定されたジョブ情報をジョブチケットとして発行することができる。コマンドボタン27は、いわゆるクリアボタンで設定されたジョブ情報をクリアする。

【0058】このように、クライアント12において、機能情報ファイルに基づいて、データ処理ステーションの機能の内容を表示することができ、さらには、ジョブ情報を簡単に設定できるようにメニュー化した機能情報を表示することができる。これは、機能情報が構造化文書であるマークアップ言語文書で表現されているからである。

【0059】なお、これらクライアント12で行われるメニュー表示処理するためのアプリケーションプログラムは、クライアントに初期の時点でインストールしてもよいし、XML文書とともにサービスプロバイダ12から配信するように構成してもよい。この場合は、アプリケーションプログラムは、java（サンマイクロシステムズ社開発）やjavascript（ネットスケープ社開発）などのオペレーティングシステムに依存しない言語で記述される。また、アプリケーションプログラムを用いる代わりに、スタイルシートと呼ばれるXML文書の形式で記述されたファイルを用いてもよい。この場合、スタイルシートをサービスプロバイダ12からダウンロードして、このシートに記述されたスタイルに基づいて表示す

#### ジョブチケット

```

1      <?xml version="1.0"?>
2
3      <!DOCTYPE XMLMenu SYSTEM
         "http://www...../XMLTicketsub.dtd">
4
5      <Ticket>
6          <Kind>RIP</Kind>
7          <Function Name="LPI">
8              <Item>400</Item>
9          </Function>
10         <Function Name="Screen">
11             <Item>FM</Item>
12         </Function>
13     </Ticket>

```

【0063】このマークアップ言語文書の1行目で、XMLで記述された文書であることを宣言し、3行目で、このマークアップ言語文書（XML文書）は、定義ファイルDFS（ファイル名「XMLMenuSub.DTD」）に基づいた文書型定義に従った文書であることを示し、5行以降で、各要素の内容を記述している。

【0064】表5で表されるジョブチケットの詳細について説明する。まず、要素「Ticket」の属性「ID」は設定されていない。しかし、参照している文書型定義（定義ファイルDFS）において、属性「ID」のデフォルト「Ticket」に設定されているので、要素「Ticket」の要素内容が、ジョブチケットであることを示している。また、要素「KIND」として「RIP」が設定されている。すなわち、この装置は、RIP装置であることを示している。さらに、「LPI」機能は、「400線」が選択され、「Screen」機能は、「FM」が選択されたことを示している。

【0065】このように、ジョブチケットおよび機能情報ファイルは、共通の文書型定義で従うXML文書で表現されている。ジョブチケットを表すのか機能情報を表すのかは、要素「Ticket」の属性「ID」が有する値により区別することができる。

【0066】なお、ジョブチケットを発行するためのプログラムは、機能メニュー表示用のプログラムと同様

る。

【0060】[(iv) ジョブチケットの発行]クライアント12では、図7で示された機能メニューに従って、データ処理内容を示すジョブ情報を設定する。すなわち、オプションボタン23および25において希望する条件を選択し、その後コマンドボタン「Submit」をクリックする。これにより、クライアント12は、XML文書で記述されたジョブチケットを発行する。

【0061】表5に、各機能において、「400線」および「FM」のジョブ情報が選択されたXML文書で記述されたジョブチケットの例を示す。

【0062】

【表5】

に、クライアントに初期の時点でインストールしてもよいし、XML文書とともにサービスプロバイダ12から配信するように構成してもよい。この場合は、アプリケーションプログラムは、java（サンマイクロシステムズ社開発）やjavascript（ネットスケープ社開発）などのOSの種類に依存しない言語で記述される。

【0067】[(v) データ処理]ジョブチケットを受け取ったサービスプロバイダ11は、機能情報ファイルの登録元のデータ処理ステーションにジョブチケットを配信してもよいし、また、データ処理において最適処理が可能なデータ処理ステーションにジョブチケットを配信してもよい。前者の場合は、各機能情報ファイルに対応するデータ処理ステーションをIPアドレスで管理しているので、該当するIPアドレスに対してジョブチケットを配信する。

【0068】後者の場合、ジョブチケットに含まれるジョブ情報を解析し、ジョブ情報として設定されたデータ処理内容が実現可能なデータ処理ステーションに登録されている機能情報ファイルから検索する。条件に該当する複数の機能情報ファイルに対応する複数のデータ処理ステーションのなかから、データ処理効率において、最適なデータ処理ステーションにジョブチケットを配信する。例えば、複数のデータ処理ステーションがデータ処理ステーションが、設定されたジョブ情報の条件

に満足する場合、現在、他のクライアントからのデータ処理を実行していないデータ処理ステーションをジョブチケットの配信先として決定する。

【0069】ジョブチケットが配信されたデータ処理ステーションは、記述されたジョブ情報に従って、データ処理を行う。なお、データ処理の対象となる画像ファイルは、ジョブチケット内に、そのファイルを参照するアドレスを記述してもよいし、ジョブチケットと一緒に別々のファイルで送るように構成してもよい。

【0070】【変形例】本実施の形態において、クライアントとデータ処理ステーション間にデータ配信管理を行うサービスプロバイダを設けたが、1つのクライアントに対して1つのデータ処理ステーションをネットワーク接続する場合は、このサービスプロバイダを設けてもよい。

【0071】また、本実施の形態において、サービスプロバイダと、データ処理ステーションは、別々のワークステーションにより構成したが、同一のワークステーションで構成してもよい。

【0072】また、本実施の形態において、文書型を定義するための文書型定義ファイルをネットワーク上で参照可能な位置に格納したが、予め設定された定義に従ってマークアップ文書を記述すれば、定義ファイルを改め

て設ける必要がない。

【図面の簡単な説明】

【図1】本発明のデータ処理ワークフローシステムの構成図である。

【図2】定義ファイルDFMで定義される文書型定義をツリー構造で表した例である。

【図3】定義ファイルDFSで定義される文書型定義をツリー構造で表した例である。

【図4】本発明のデータ処理ワークフローの概略の流れを説明するための図である。

【図5】機能情報ファイルの内容をツリー構造で表した例である。

【図6】RIP装置の機能情報ファイルの内容をツリー構造で表した例である。

【図7】クライアントのモニタに表示される機能メニューの表示例である。

【符号の説明】

1…データ処理ワークフローシステム

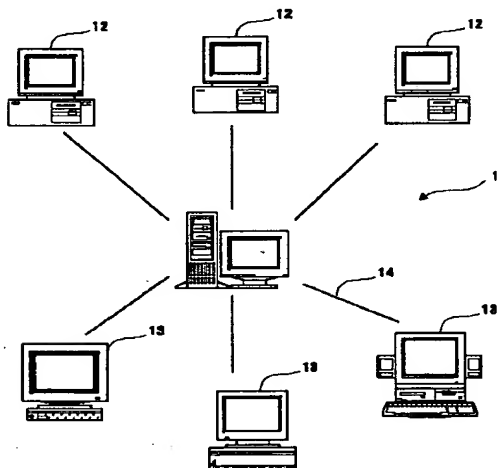
11…サービスプロバイダ

12…クライアント

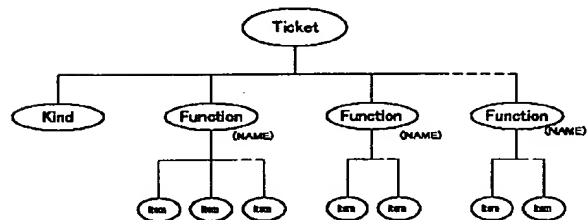
13…データ処理ステーション

14…ネットワーク

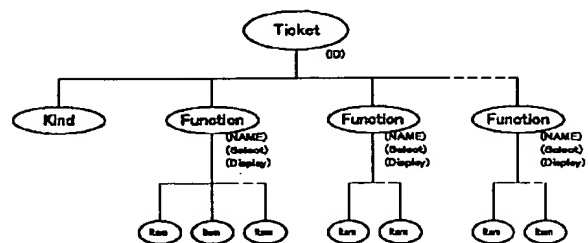
【図1】



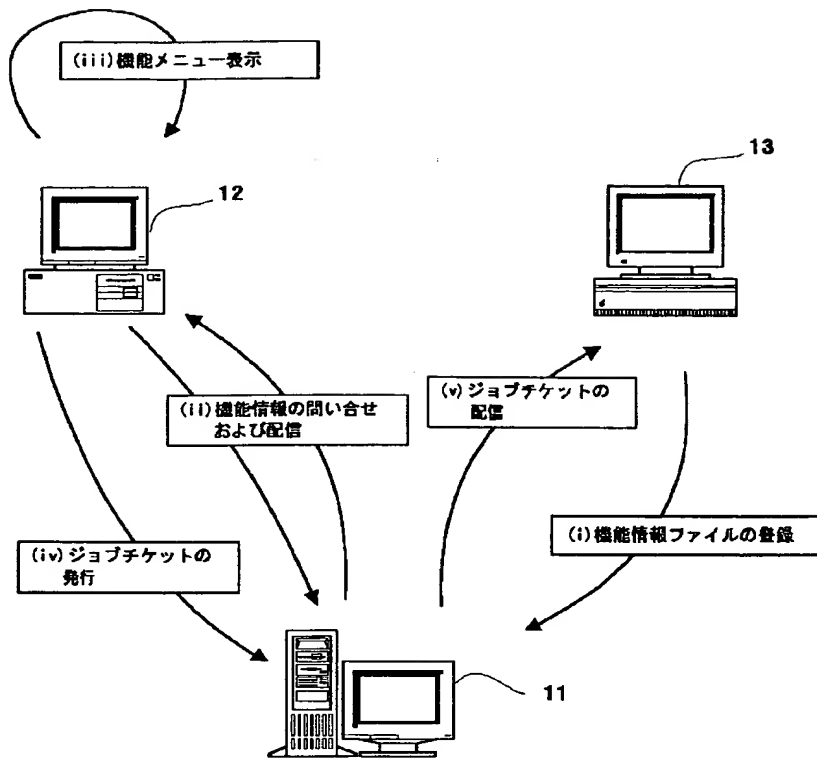
【図2】



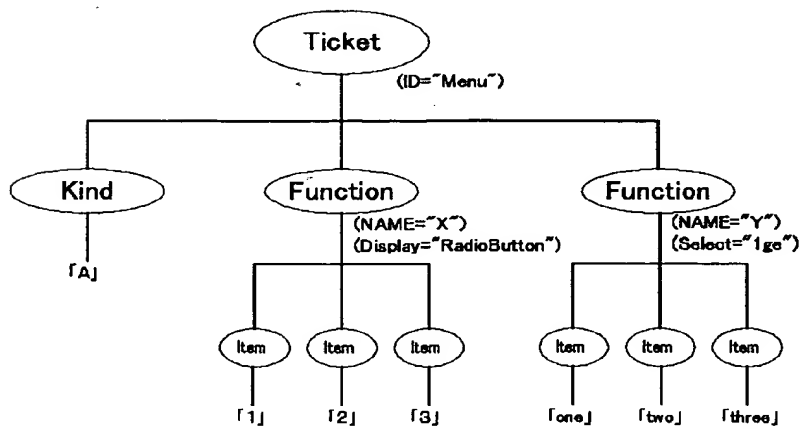
【図3】



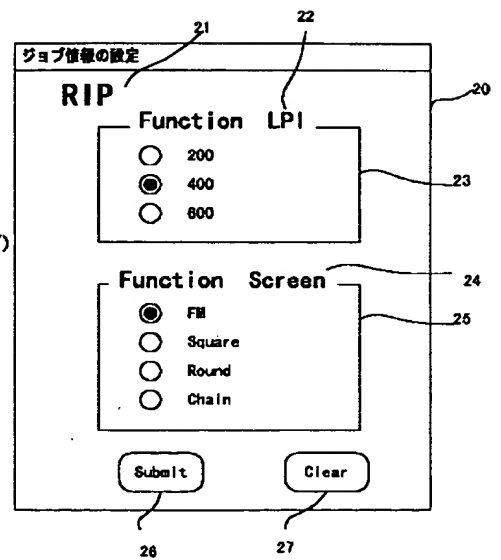
【図4】



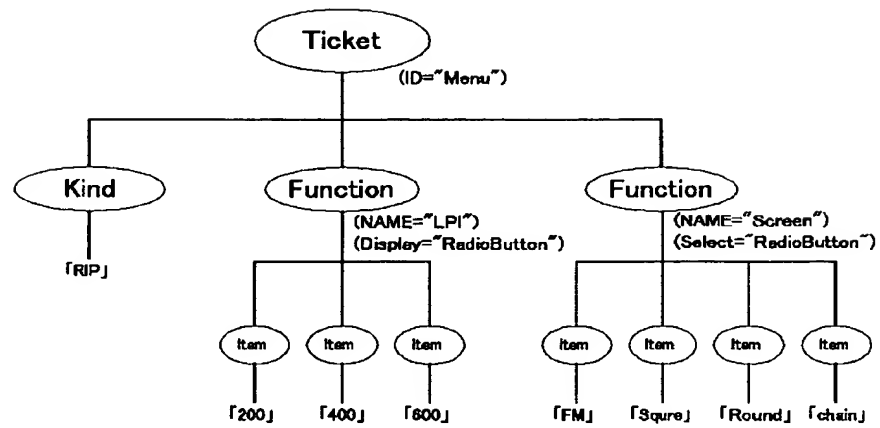
【図5】



【図7】



【図6】



フロントページの続き

(72)発明者 西村 晃  
京都市上京区堀川通寺之内上る4丁目天神  
北町1番地の1 大日本スクリーン製造株  
式会社内